

Livraisons
d'Histoire
de l'Architecture

Livraisons de l'histoire de l'architecture

32 | 2016

Les représentations de l'architecture

Dessiner avec l'ordinateur dans les années soixante : le design et ses pratiques à l'aube de l'ère numérique

Drawing with computers in the 1960s: Rethinking design at the dawn of the digital age

Computerzeichnungen in den 1960er Jahren. Design und seine Praxis am Beginn des digitalen Zeitalters

Jordan Kauffman

Traducteur : Marie-Luce Kauffman



Édition électronique

URL : <http://journals.openedition.org/lha/643>

DOI : 10.4000/lha.643

ISSN : 1960-5994

Éditeur

Association Livraisons d'histoire de l'architecture - LHA

Édition imprimée

Date de publication : 31 décembre 2016

Pagination : 105-123

ISSN : 1627-4970

Référence électronique

Jordan Kauffman, « Dessiner avec l'ordinateur dans les années soixante : le design et ses pratiques à l'aube de l'ère numérique », *Livraisons de l'histoire de l'architecture* [En ligne], 32 | 2016, mis en ligne le 31 décembre 2018, consulté le 19 avril 2019. URL : <http://journals.openedition.org/lha/643> ; DOI : 10.4000/lha.643

Tous droits réservés à l'Association LHA

Par Jordan KAUFFMAN

DESSINER AVEC L'ORDINATEUR DANS LES ANNÉES SOIXANTE : LE DESIGN ET SES PRATIQUES À L'AUBE DE L'ÈRE NUMÉRIQUE

Les premières explorations de l'utilisation des ordinateurs dans le domaine du design¹ ont débuté dans les années 1960, près de trente ans avant que le « tournant numérique » (*digital turn*) ne s'empare du champ de la production architecturale. À cette période, ces machines et leur rôle dans le processus de conception suscitent beaucoup d'enthousiasme, notamment dans les professions pour lesquelles dessiner est primordial, et chez qui l'introduction d'une nouvelle technologie semble offrir un réel potentiel de changement vis-à-vis des pratiques du dessin.

Cet optimisme d'alors, de même que l'omniprésence des ordinateurs dans le design aujourd'hui, a toutefois eu tendance à occulter les nombreuses difficultés auxquelles les outils informatiques ont dû faire face afin d'intégrer les métiers de la création. À leurs débuts, les ordinateurs sont en effet bien plus facilement assimilés par les professions vouées à l'analyse de nombres ou à la résolution d'équations, telles que l'ingénierie et l'économie. Ils sont d'ailleurs originellement développés par et pour des ingénieurs et des économistes pour répondre plus spécialement à ces besoins².

L'auteur souhaite remercier Marie-Luce Kauffman pour la traduction en français de cet article.

1. Alors qu'en français *design* est généralement utilisé pour qualifier une activité de création de produits et concepts à visée industrielle ou commerciale, le mot prend dans cet article une acception plus large, suivant l'usage commun en anglais qui recouvre la production de l'ensemble des professions et pratiques créatives, incluant notamment l'architecture.
2. Le mot *computer* lui-même était employé, avant l'invention de la machine du même nom, pour désigner un individu affecté spécifiquement à faire des calculs. Le glissement de sens de la personne à la machine, tel qu'il est utilisé aujourd'hui en anglais, est significatif : les ordinateurs étaient à l'origine développés pour effectuer les tâches réalisées par ces individus, plus rapidement et avec moins d'erreurs. On le voit notamment avec l'invention de la machine de différence de Charles Babbage (calculatrice mécanique inventée pour compiler des tables mathématiques et présentée en 1821) et la tabulatrice électrique d'Herman Hollerith (système conçu pour traiter les données du recensement américain de 1890), mais aussi avec la machine analytique du même Babbage (une machine élaborée en 1834 pour être en mesure de réaliser tout type de calcul mathématique), Harvard Mark I (premier ordinateur automatique, livré en 1944), et l'ENIAC (premier ordinateur entièrement numérique en 1947). Pour une histoire approfondie du développement des ordinateurs, voir l'ouvrage de Martin Campbell-Kelly et William Aspay, *Computer : A History of the Information Machine*, New York, Basic Books, 1996.

En design, et plus particulièrement en architecture, les ordinateurs restent alors surtout cantonnés aux petites tâches quotidiennes des agences, telles que la gestion des fichiers administratifs. Ils sont néanmoins perçus comme aptes à réaliser des calculs essentiels à la conception d'ouvrages et, avec suffisamment d'éléments à leur disposition, capables de générer des plannings. On imagine même que dans un futur plus ou moins proche, ils vont libérer les architectes de leurs tâches les plus prosaïques en leur permettant de se consacrer davantage aux aspects créatifs de leur métier³.

Rétrospectivement, c'est vis-à-vis de la création elle-même que le passage à l'informatique semble avoir été le plus compliqué. Parce que le processus créatif n'est pas aisément décomposable en règles systématiques, scientifiques ou mathématiques pouvant générer des réponses exactes, parce que pour designers et architectes, l'expression d'une idée à travers le dessin est le mode principal de création et de communication, il a fallu à l'époque s'efforcer de concilier ordinateurs et pratiques du dessin. Cet article explore précisément cette question : comment, au stade précoce de la conception assistée par ordinateur, la machine a affecté et assimilé l'acte de dessiner, le processus de dessin et de création, et les dessins eux-mêmes.

Interfaces graphiques et pratiques du dessin

Le début des années 1960 a ceci de remarquable que les ordinateurs ne sont pas encore des machines standardisées. En ce temps-là, il n'existe pas d'interfaces uniformes, et encore moins de matériel ou de logiciel standard. Il n'y a pas non plus de manière prédéfinie et normative d'interagir avec ces machines. Les ordinateurs sont élaborés suivant des technologies différentes pour répondre au mieux aux objectifs de ceux qui les construisent et qui écrivent leurs programmes. Ils accomplissent des tâches qui diffèrent d'un ordinateur à un autre, chacun impliquant en contrepartie des limites qui lui sont propres. C'est précisément à cette période que l'on voit apparaître les tout premiers systèmes informatiques à interface graphique : parmi eux, EDM et Digigraphics, développés par Itek, DAC-1, conçu par la firme automobile General Motors, et Sketchpad, créé par Ivan Sutherland pour sa thèse de doctorat au Massachusetts Institute of Technology (MIT), vont s'avérer particulièrement importants pour leurs avancées technologiques vis-à-vis du rôle de la machine dans le processus de conception.

Préalablement au développement de ces systèmes pionniers a lieu l'avènement de l'informatique numérique à grande échelle. L'une des innovations majeures de l'après-guerre est le Whirlwind, élaboré à partir de 1946 au MIT, d'abord au sein du Servomechanisms Laboratory, puis au Digital Computer Laboratory, et enfin à

3. Voir plus particulièrement Vladimir Bazjanac, « The Promises and the Disappointments of Computer-Aided Design », dans Nicholas Negroponte (dir.), *Reflections on Computer Aids to Design and Architecture*, New York, Petrocelli/Charter, 1975, p. 17-26.



Ill. 1 : Un terminal du Semi Automatic Ground Environment (SAGE), c. 1958. © Photographie reproduite avec l'aimable autorisation du MIT Lincoln Laboratory, Lexington, Massachusetts.

la Digital Computer Division du Lincoln Laboratory. Premier ordinateur numérique à haute vitesse, le Whirlwind va servir de socle au programme gouvernemental américain *Semi-Automatic Ground Environment* (SAGE), développé également au Lincoln Laboratory dès 1950⁴. SAGE est un système de radar intégré qui utilise des ordinateurs pour centraliser et collecter de grandes quantités d'informations issues des stations radar à travers les États-Unis, donnant une image complète de l'espace aérien national. Plus important encore, c'est le premier système avec lequel il est possible d'interagir avec des graphismes apparaissant sur un écran. Un écran circulaire à tube cathodique de 24 pouces de diamètre (soit un peu plus de 60 centimètres) affiche les informations du trafic aérien. Un opérateur utilise un pistolet optique (*light gun*) et actionne les divers commutateurs et boutons de sa console pour sélectionner des cibles sur l'écran afin de recueillir davantage d'information, choisir la défense appropriée, et donner l'ordre d'attaquer des sites locaux (ill. 1). Il y a communication en temps réel entre l'utilisateur et les informations données par l'ordinateur. C'est alors la démonstration qu'une interface interactive comportant des représentations graphiques (et non pas seulement des mots ou du texte) peut être conçue, mise en œuvre et utilisée efficacement⁵.

4. Le Whirlwind, comme de nombreuses technologies informatiques des débuts, a été mis au point grâce aux financements militaires et gouvernementaux américains. Voir Jay W. Forrester et Robert R. Everett, « The Whirlwind Computer Project », *IEEE Transactions on Aerospace and Electronic Systems*, septembre 1990, v. 26, n° 5, p. 903-910 ; pour SAGE, voir également Kent C. Redmond et Thomas M. Smith, *From Whirlwind to MITRE : The R+D Story of the SAGE Air Defense Computer*, Cambridge, MA : MIT Press, 2000, p. 1-20.

5. Alors que l'opérateur est assis face à une petite borne, l'ordinateur nécessaire au traitement de toutes les informations est installé dans une pièce séparée. Établi sur vingt-quatre sites différents reliés entre eux par des lignes téléphoniques, SAGE exige sur chaque site près de 2 000 mètres carrés d'espace disponible pour accueillir une zone de consoles et deux ordinateurs. Considéré comme le plus grand, le plus lourd et le plus coûteux ordinateur jamais construit, il demandait plus de cent personnes pour le faire fonctionner et possédait ses propres circuits de refroidissement. Voir Martin Campbell-Kelly et William Aspay, *op. cit.*, p. 167-168.

Néanmoins, l'applicabilité d'un tel système au processus de conception est extrêmement limitée. L'opérateur n'a qu'une influence minimale sur ce qui est affiché à l'écran, et ne peut rien modifier en fonction de ses besoins, pas plus que d'importer des informations ou éditer des commandes. Il ne peut ni entrer ses propres données, ni changer fondamentalement celles qu'il voit à l'écran. Il va falloir attendre la dissémination dans l'industrie des technologies développées pour SAGE et la reconversion dans le secteur privé de ceux qui ont œuvré au développement de ce programme militaire ultrasecret pour qu'apparaissent de nouvelles expérimentations avec des interfaces graphiques.

EDM et Digigraphics

En 1959, Jack Gilmore, Charles Adams et Norm Taylor – tous trois ayant travaillé sur le projet SAGE – proposent d'élaborer un système d'interface graphique au service des ingénieurs. Gilmore et Adams sont alors associés chez Charles W. Adams Associates, une jeune société de programmation informatique, et Taylor travaille à Itek Corporation, un entrepreneur au service de la Défense américaine. En 1960, Itek offre de financer leur projet tout en chargeant Adams Associates de la programmation du nouveau système. Ainsi naît l'*Electronic Drafting Machine*, ou EDM⁶, qui sera parmi les tous premiers dispositifs de dessin assisté par ordinateur (DAO).

EDM apporte une solution à un problème qui empêchait jusque-là la technologie informatique d'investir les champs du design : le manque de communication entre les utilisateurs et les ordinateurs. En 1962, ainsi que le souligne un court article du magazine *Time* présentant EDM et intitulé « Beating the Language Barrier » (vaincre la barrière de la langue), l'un des obstacles majeurs à l'utilisation de l'ordinateur en design est qu'un « programmeur qualifié doit souvent passer plusieurs jours à réduire les éléments d'un problème à un code numérique ou électronique avant même de pouvoir tenir une brève conversation avec sa machine⁷ ». La tâche de programmeur est bien trop fastidieuse pour une profession qui ne pense pas son travail de manière similaire, si décomposée et abstraite. Les designers ont déjà une langue qui leur est propre, grâce à laquelle ils communiquent et développent leurs idées : le dessin. Si des machines sont élaborées pour soutenir utilement le processus de conception, il faut alors pouvoir les utiliser sans que la connaissance d'un langage programmatique ne soit préalablement requise⁸.

6. Alors que ce système s'est fait connaître sous le nom d'*Electronic Drafting Machine*, il semble qu'il ait eu au tout début une certaine confusion sur la signification de l'acronyme EDM. Dans une description sommaire du produit en 1962, le magazine *Time* précise en effet que plusieurs dirigeants d'Itek pensent alors qu'EDM correspond à *Engineering Design Machine* ou à *Engineering Drafting Machine*. Voir l'article « Beating the Language Barrier », *Time*, 2 mars 1962, p. 74-75.

7. *Ibid.*, p. 74 : « [...] [A] skilled programmer must often spend days reducing the elements of a problem to numerical or electronic code before he can hold even a brief conversation with his machine ».

8. On peut cependant voir quelques années après que certains pensent qu'architectes et autres designers auront malgré tout besoin d'apprendre le langage informatique afin d'être en mesure d'utiliser l'ordinateur le plus efficacement possible. Voir par exemple John Eberhard, « Panel Discussion :



Ill. 2 : Le système Digigraphics opéré par la Control Data Corporation, c. 1963. © Photographie reproduite avec l'aimable autorisation de Charles Babbage Institute Archives, University of Minnesota Libraries.

EDM se présente comme une étape vers la résolution de ce problème. Au lieu de devoir réduire les problèmes de design à des équations, le système à la possibilité de traiter directement les données graphiques entrées par un utilisateur. Il utilise pour ce faire une technologie basée sur le pistolet optique, appelée crayon optique (*light pen*) ou stylet, qui permet de dessiner sur un écran (ill. 2). Un bouton sur le stylet, lorsqu'il est enfoncé, signale à l'ordinateur où le tracé d'une ligne commence et prend fin. Les données sont stockées dans la mémoire de la machine et peuvent être rappelées à tout moment, à des échelles différentes. Les « dessins » – des lignes lumineuses sur un moniteur – sont ainsi modifiables.

The Impact of Computers on Architecture », Murray Milne (dir.), *Computer Graphics in Architecture and Design : proceedings of the Yale Conference on Computer Graphics in Architecture, held in New Haven, Connecticut, April 1968*, New Haven, Yale School of Art and Architecture, 1969, p. 55-59.

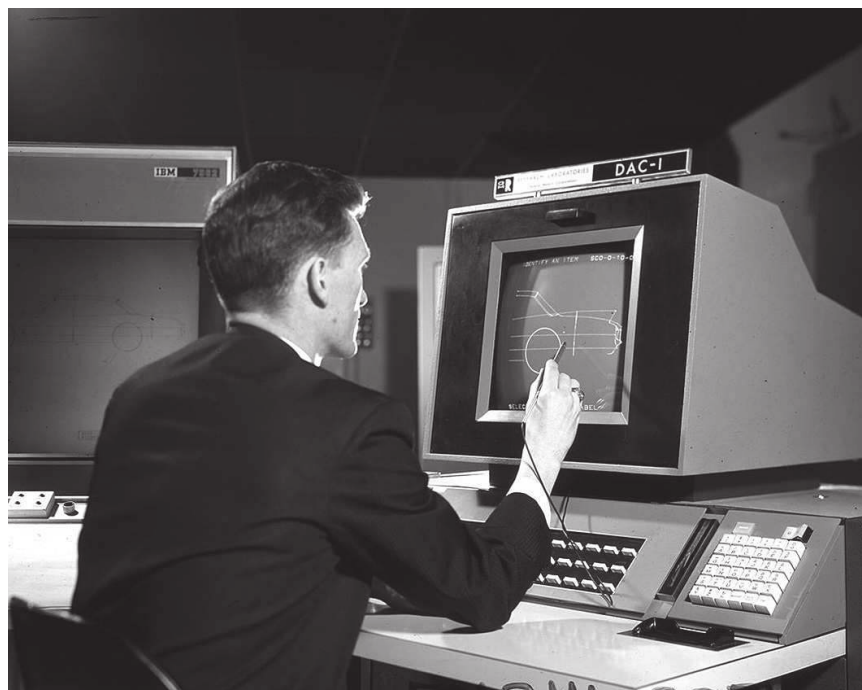
Itek vend un exemplaire du système à l'Armée de l'air américaine, qui rebaptise le programme d'interface Digigraphics Display Program, avant que l'une des plus importantes compagnies de superordinateurs des années 1960, la Control Data Corporation (CDC), ne rachète l'ensemble et ne l'exploite sous le nom de Digigraphics. Tout premier système de DAO à être mis sur le marché, Digigraphics coûte environ \$500 000 à l'achat, et \$2 000-3 000 par mois à la location. Compte tenu de ces prix prohibitifs, seuls Lockheed Corporation et Martin Marietta, deux constructeurs aéronautiques, ainsi que la Marine américaine, s'en portent acquéreurs⁹. Aucune agence d'architecture ou de design n'a pour sa part les moyens d'acheter le programme. Ne se révélant ni vraiment commercialisable, ni rentable, ce projet est abandonné après seulement quelques années d'existence.

General Motors et DAC-1

Dans les mêmes années, un autre système de conception graphique est mis au point par le constructeur automobile américain General Motors (GM) pour améliorer le processus de création de ses voitures. Connu sous le nom de *Design Augmented by Computer*, ou DAC-1¹⁰, ce système est officiellement présenté lors de la *Joint Computer Conference* à l'automne 1964.

GM a commencé dès 1952 à faire usage des ordinateurs pour l'évaluation scientifique de ses produits et ce, grâce à un système de cartes perforées. L'intérêt grandissant porté aux technologies informatiques conduit le constructeur automobile à créer à la fin des années 1950 le Data Processing Department, une unité spécialisée aux seins de ses laboratoires de recherche. En 1959, ce département procède à la première démonstration de faisabilité d'un programme informatique capable de gérer les entrées graphiques. Bien que modeste, celle-ci fait la preuve que « des lignes tracées sur un film peuvent être scannées et numérisées sous le contrôle des programmes d'ordinateur¹¹ ». Encouragé par ces possibilités, GM décide de poursuivre son étude des interactions graphiques entre utilisateur et ordinateur à travers le développement de ce nouveau programme, en association avec la firme IBM. Les objectifs déclarés de DAC-1 sont de « permettre une communication graphique conversationnelle entre l'homme et la machine¹² », et se résument en quatre points :

9. « Beating the Language Barrier », *op. cit.*, p. 75. En plus des prix à l'achat et à la location, l'article liste ainsi ceux qui se montrent intéressés par EDM : « urban developers looking for better ways to plan cities, railroaders seeking quicker ways to unsnarl traffic, aerospace engineers who hope for swifter and finer design work ».
10. Initialement, DAC-1 s'appelle « Digital Design », mais son nom est changé pour éviter toute confusion sur l'emploi du mot *digital*. On craint en effet que les gens ne s'imaginent que le système est utilisé pour concevoir des chiffres (*digits* en anglais) ou se réfère aux doigts de la main. Pour une analyse plus complète, voir Fred N. Krull, « The Origin of Computer Graphics within General Motors », *IEEE Annals of the History of Computing*, automne 1994, v. 16, n° 3, p. 40-56.
11. Voir le communiqué de presse produit par General Motors, *Background information on GM Group Responsible for Development of DAC-1 : Design Augmented by Computers*, 28 octobre 1964, p. 1.
12. *Ibid.*, p. 2 : « to permit conversational man-machine graphical communication ».



Ill. 3 : Le système Design Augmented by Computers (DAC-1) élaboré par General Motors, c. 1964.
© Photographie reproduite avec l'aimable autorisation de General Motors.

1) le système informatique doit être en mesure de lire des dessins existants et de les numériser ; 2) des éléments de commande sont nécessaires pour manipuler et modifier les dessins une fois numérisés ; 3) les dessins produits sur ordinateur ont besoin d'être sauvegardés pour que l'on puisse s'y référer ultérieurement ; et 4) le système doit également permettre l'intégration d'informations textuelles, en plus des informations graphiques¹³. Par cela, GM reconnaît la possibilité d'un processus de conception dans lequel ordinateur et designer s'engagent l'un l'autre dans un dialogue graphique.

DAC-1, à l'instar d'EDM, est capable de retracer des lignes dessinées sur un écran d'ordinateur. Il est décrit comme étant « similaire au dessin des lignes d'un crayon sur l'écran cathodique [d'une] télévision¹⁴ ». Cependant, contrairement à EDM, le crayon optique est utilisé non pas directement sur l'écran mais sur une surface conductrice placée sur le moniteur. Le système détecte l'emplacement du stylet par le contact que celui-ci produit sur la surface, qui lui transmet alors l'information (ill. 3). C'est donc un système qui se rapproche davantage du dessin avec un stylo ou un crayon que de l'utilisation d'un crayon optique, car le contact est ici nécessaire pour que le système détecte l'entrée. À terme, DAC-1 va toutefois se montrer impraticable à l'usage. L'effort qu'il faut pour maintenir un bras tendu

13. Edwin L. Jacks, « A Laboratory for the Study of Graphical Man-Machine Communications », in American Federation of Information Processing Societies, *AFIPS Proceedings of the October 27-29, 1964 Fall Joint Computer Conference, part I*, 1964, p. 343-350.

14. Voir le communiqué de presse produit par General Motors, *op. cit.*, p. 5.

pendant de longues périodes s'avère trop pénible pour les designers, voire impensable dans leur pratique quotidienne, pour pouvoir être utilisé efficacement. Le succès du système s'oppose aux limitations physiques de ses opérateurs et des normes du dessin dans la pratique du design, où l'on dessine en générale sur une surface plus horizontale.

Sketchpad

Un troisième système de conception graphique interactif, Sketchpad, est développé au MIT par Ivan Sutherland pour la thèse de doctorat qu'il présente en 1963¹⁵ (ill. 4). Ce système informatique est créé au moment où la conception assistée par ordinateur (CAO) commence à être théorisée au MIT. En décembre 1959, afin d'explorer les possibilités du « Computer-Aided Design », des membres des départements de génie mécanique et de génie électrique se sont ainsi associés au sein d'un projet financé par l'Armée de l'air américaine. À l'issue d'une étude de plusieurs mois, ils reconnaissent que l'ordinateur va et doit influencer d'une certaine façon sur le processus de conception. La CAO, écrivent-ils, n'est « PAS un miroir de la pratique actuelle du design... Quelle que puisse être sa signification ultime, elle ne s'apparente pas à notre vision actuelle du design telle qu'il est pratiqué aujourd'hui¹⁶ ». Bien que ces pratiques puissent évoluer, ces chercheurs insistent sur le fait que l'ordinateur ne devrait pas se voir attribuer un trop grand pouvoir de décision. Ils préfèrent plutôt envisager un système dans lequel l'homme et la machine fonctionneraient en tandem avec la même capacité de conception¹⁷.

Sketchpad marque une étape importante dans la concrétisation de ces idées, même s'il n'en est pas explicitement issu. Comme pour l'EDM et DAC-1, son utilisateur trace directement des lignes sur un écran vertical à l'aide d'un stylet et actionne des boutons d'entrées afin de définir des contraintes. Son interface se montre toutefois bien plus interactive que celle des deux autres systèmes. Son programme accepte ainsi l'usage des points, des segments de ligne, et des arcs en tant qu'éléments de base, mais permet surtout que ceux-ci soient enregistrés dans les dessins de référence, qui peuvent alors être copiés ou instanciés. En outre, Sketchpad gère non seulement les contraintes explicites, ajoutées à des entités déjà dessinées, mais aussi les contraintes implicites, créées au fur et mesure du tracé des entités. Par exemple, si l'on dessine une ligne et que l'on approche le curseur près du point d'extrémité d'une ligne existante, le curseur rejoint automatiquement cette extré-

15. Ivan E. Sutherland, *Sketchpad: A man-machine graphical communication system*, Ph.D. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering, 1963, 176 p.

16. Douglas T. Ross, *Computer-Aided Design: A Statement of Objectives*, technical memorandum 8436-TM-4, Cambridge, MA, Massachusetts Institute of Technology, Department of Electrical Engineering, Electronic Systems Laboratory, septembre 1960, p. iii.

17. *Ibid.* : « The objective of the Computer-Aided Design Project is to evolve a man-machine system which will permit the human designer and the computer to work together on creative design problems ».



Ill. 4 : Le Sketchpad d'Ivan Sutherland, c. 1963. © Photographie reproduite avec l'aimable autorisation du MIT Museum.

mité. Celui-là même se souvient par la suite que les deux lignes sont connectées, et si l'on bouge une ligne, l'autre ligne se déplace avec elle.

Sketchpad inclut dix-sept formes de contraintes, y compris verticale, horizontale, perpendiculaire, simultanée, parallèle, alignée, et de taille égale, qui peuvent être combinées pour créer des relations plus complexes. Le programme permet leur affichage visuel à l'écran, utilisant une icône pour chaque type. Avec un système de contraintes aussi sophistiqué, il est possible d'esquisser vaguement une forme et d'y greffer des relations pour la modifier jusqu'à la forme exacte recherchée. Le dispositif autorise également le tracé asservi à un point fixe (*rubberbanding*) lors de l'élaboration ou la correction des entités, de sorte que les entités s'étirent suivant les déplacements du curseur. Il offre de surcroît la possibilité d'utiliser des mouvements, des rotations et des échelles dynamiques, toujours liés aux déplacements du curseur, ainsi que celle de zoomer ou de travailler sur un format panoramique. Ces fonctionnalités restent constamment accessibles, que cela soit pendant le dessin ou la correction.

En repensant à son projet quelques années après, Sutherland écrit : « Un dessinateur ordinaire est indifférent à la structure de son matériel de dessin. La plume et l'encre ou un crayon et du papier n'ont aucune structure inhérente. Ils ne font que des salissures sur le papier. Le dessinateur est principalement soucieux des dessins en tant que représentation du processus de création. Le comportement d'un dessin réalisé par ordinateur, quant à lui, dépend essentiellement de la structure topologique et géométrique construite dans la mémoire de l'ordinateur comme le résultat de gestes dessinant¹⁸ ». Bien qu'il soit inexact d'affirmer que les matériaux qui structurent le dessin manuel n'ont pas d'effet sur ce qui est produit¹⁹, Sutherland souhaite ici souligner qu'il existe des définitions et des relations entre les lignes et les éléments d'un dessin créé par ordinateur qui sont absentes des dessins réalisés sur papier. Chaque élément tracé sur l'écran, codé avec des informations qui ne sont pas présentes dans un dessin manuel, possède ainsi des caractéristiques qui lui sont propres, déterminant sa nature et la façon dont il se comporte et se rapporte à d'autres éléments du dessin. Malgré de réels avantages, ceci va néanmoins représenter un obstacle à la réussite de la version initiale de Sketchpad. Dans celle-ci en effet, les relations établies entre les éléments, même aussi simples que la jointure des extrémités de deux lignes, ne peuvent être annulées. Ces relations sont permanentes et les défaire nécessite de refaire l'ensemble du dessin. En outre, Sketchpad est incapable de gérer des dessins complexes et manque rapidement de mémoire.

Ceci souligne bien qu'à la fois matériel et logiciels influent sur le résultat d'une création réalisée par ordinateur. Alors que du point de vue de ceux qui soutiennent le développement de cette nouvelle technologie, Sketchpad représente une voie

18. Ivan Sutherland, « Structure in Drawings and the Hidden-Surface Problem », dans Nicholas Negroponte (dir.), *Reflections on Computer Aids to Design and Architecture*, New York, Petrocelli/Charter, 1975, p. 75.

19. Sutherland se montre peu critique envers l'influence que les matériaux de dessin ont sur le dessin lui-même ; il suffit pourtant de se rappeler comment parchemin, stylets et encres ont donné lieu à certaines caractéristiques linéaires de l'architecture gothique, ou comment le papier, à travers l'esquisse, a institutionnalisé la pratique du dessin.

d'exploration très intéressante, ces possibilités constituent pour d'autres des limitations de la souplesse inhérente au dessin sur papier. Elles ne laissent, par exemple, pas de place à l'ambiguïté, l'une des qualités du dessin qui permet le plus de recherche créative.

EDM, DAC-1 et Sketchpad incarnent au début des années 1960 les avancées les plus significatives dans le domaine des procédés graphiques et de la conception assistée par ordinateur. Pour chacun d'eux, le dessin est choisi comme moyen effectif de communiquer entre l'homme et la machine, car il le médium par lequel sont transmises les intentions des designers et à travers lequel se développent de nouveaux projets. Cependant, les limitations de la technologie forment un obstacle difficile à franchir. Malgré l'utilisation du crayon optique qui signale à l'ordinateur la présence de données, ou celle d'un écran conducteur qui fournit tout du moins une rétroaction tactile similaire, aucun de ces premiers systèmes ne peut reproduire les qualités ni atteindre les possibilités du dessin manuel. À défaut de retranscrire la fluidité présente dans l'acte de dessiner, ils encouragent toutefois à penser les dessins en termes d'éléments individuels et relationnels.

L'Architecture Machine Group

En 1967, quatre ans après que Sutherland ait présenté Sketchpad, un groupe composé d'architectes, d'ingénieurs électriques et de leurs étudiants prend forme au sein de l'École d'architecture du MIT. Fondé par Nicholas Negroponte et Leon Grossier avec l'idée de pousser les technologies du dessin informatique au-delà de leurs limites initiales, l'Architecture Machine Group (Arch Mac) se donne pour objectif d'enquêter sur l'utilisation des ordinateurs dans la conception architecturale. C'est le seul groupe qui, dans un environnement de recherche architecturale, va spécifiquement traiter de cette question²⁰ (ill. 5).

Arch Mac considère que des systèmes informatiques tels que Sketchpad, DAC-1 et l'EDM sont inintelligents, car la personne qui les alimente en données en possède le contrôle total²¹ : l'ordinateur, au fond, se plie à la volonté du designer pour représenter simplement ce que celui-ci a à l'esprit ; de même, l'opérateur dessine seulement sur ordinateur. Du point de vue du groupe, si les ordinateurs sont conçus pour aider l'architecture, ils doivent alors être suffisamment intelligents pour initier un dialogue avec un designer de la manière la plus proche possible de

20. L'Architecture Machine Group reçoit l'appui de Lawrence Anderson, doyen de l'École d'architecture et de planification du MIT, qui voit en cette initiative un moyen de faire évoluer la pédagogie architecturale du MIT au-delà des enseignements Beaux-Arts résiduels ayant encore une certaine influence à l'époque. Considéré comme une unité académique jusqu'en 1980, le groupe inspirera la création de l'actuel Media Lab au MIT.

21. Ils vilipendent principalement le Sketchpad de Sutherland. Voir Nicholas Negroponte, *Soft Architecture Machines* Cambridge, MA, MIT Press, 1975, p. 57-60 et 64-65.



Ill. 5 : Les ordinateurs de l'Architecture Machine Group, 1973. © Photographie reproduite avec l'aimable autorisation du MIT Museum.

celle d'un dialogue de personne à personne²². En outre, ils doivent apprendre et évoluer au fur et à mesure de leur expérience des processus de conception, chaque machine s'individualisant de plus en plus en réponse aux situations pour lesquelles elle est utilisée. Cela signifie un système interactif dans lequel l'ordinateur prend la place d'une autre personne. Negroponte émet ainsi l'hypothèse d'une « machine assistante, [d']un être humain de substitution ». Il ajoute : « Laissez-nous construire des machines capables d'apprendre, de rechercher et de tâtonner, des machines qui seront des partenaires d'architecture, des machines d'architecture²³ ».

Tous les projets sur lesquels Arch Mac est amené à travailler sont centrés sur cette vision. Le projet le plus connu, Urban 5, est un programme informatique capable de générer des solutions aux problèmes de design urbain, tout en affichant les résultats sur un écran. Malgré la présence d'éléments graphiques, Urban 5 manipule principalement des cubes et les résultats doivent être clarifiés via l'interaction

22. Nicholas Negroponte, *The Architecture Machine*, Cambridge, MA, MIT Press, 1970, p. 17 : « No matter how fancy and sophisticated the computer graphics system, it is only a glorified blackboard or piece of paper (even though possibly three-dimensional), that is, until it overtly "talks back" and actually participates in the dialogue. »

23. *Ibid.*, p. 121. Alors que généralement Negroponte épouse la cause du dialogue concepteur-machine, ses intentions apparaissent parfois plus extrêmes, notamment lorsqu'il évoque qu'un architecte n'est même plus nécessaire pour concevoir de l'architecture. Voir ainsi Nicholas Negroponte, *Soft Architecture Machines*, *op. cit.*, p. 1 : « The general assumption is that in most cases the architect is an unnecessary and cumbersome (and even detrimental) middleman... »

du langage. À terme, ceci se révèle trop imprécis et trop lent pour ses concepteurs, qui découvrent que le langage parlé est trop pesant pour qu'humains et ordinateurs puissent communiquer efficacement lors du processus de conception. L'obligation de recourir à ces imitations d'interactions langagières conduit à l'abandon du projet après seulement quelques années.

Alors qu'ils poursuivent le développement d'autres projets, les membres du groupe prennent conscience que le dessin, qui est le mode de communication privilégié dans les métiers du design, est l'outil communicationnel-clé qui va permettre aux hommes d'interagir avec les ordinateurs. Ces derniers doivent donc apprendre à lire et à interpréter (déterminer la signification) des dessins. Pour faciliter cela, Arch Mac décide d'utiliser une tablette graphique, périphérique d'entrée se rapprochant tout particulièrement des pratiques du dessin avec un crayon et du papier. Une tablette repose généralement à plat sur une table, couverte d'une feuille de papier, et présente un réseau de fils électriques situé en dessous de la surface de support, à travers lequel sont envoyées des impulsions dont l'intensité est mesurée par un oscilloscope. Elle fonctionne avec un stylet spécial qui a la taille, le poids et l'apparence d'un stylo à plume. Sa pointe est capable de ressentir une pression similaire à celle que l'on exerce lorsque l'on utilise un crayon, l'idée étant que ce système soit le plus naturel possible pour ceux habitués à dessiner sur papier.

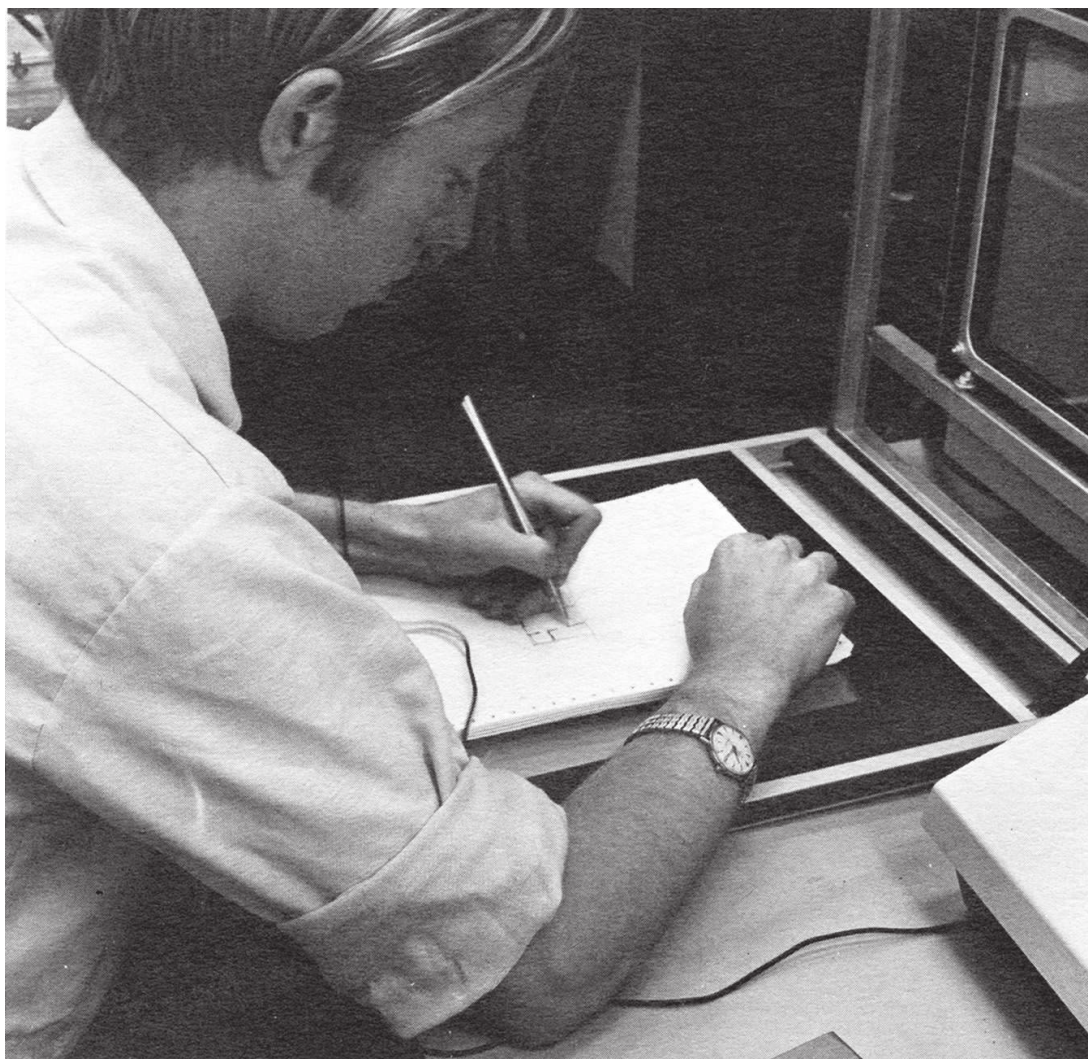
Arch Mac reçoit sa première tablette Sylvania au printemps 1970 et se lance dans le développement d'un système de reconnaissance d'esquisses (ill. 6). Les esquisses sont choisies car, selon le groupe, elles représentent le principal mode de communication grâce auquel un designer communique tant avec lui-même qu'avec les autres²⁴. Ce système est dénommé « Hunch » (intuition, en français)²⁵, car l'objectif est qu'il puisse interpréter ce qu'un designer tente de dessiner, de déterminer la signification de son esquisse, et de compléter voire achever ses intentions. Pour ce faire, il enregistre chaque bosse et imperfection présentes sur un dessin manuel, afin de « permettre à l'utilisateur d'être aussi graphiquement autonome, équivoque, et inexact qu'il ne le serait avec un partenaire humain²⁶ ». Hunch, plus que tout autre programme à ce stade, reconnaît qu'un dessin est non seulement le résultat final, mais aussi l'incarnation de tout un processus inhérent à sa création. Pour cela, le groupe développe un logiciel qui suit la vitesse, la pression et les accélérations du crayon. Il capte les données 200 fois par seconde, et peut projeter à l'écran une image numérique décrite comme « un morceau de papier de substitution²⁷ ». Le programme enregistre le dessin sur la durée, sur la base de la vitesse à laquelle il est produit, et peut redessiner en temps réel des lignes qui ont été tracées.

24. D'après Negroponte, « l'esquisse peut être considérée à la fois comme une forme de communication avec soi-même (introspection) et une forme de communication avec les autres (présentation) ». Voir Nicholas Negroponte, *Soft Architecture Machines*, op. cit., p. 64.

25. Pour une description détaillée des fonctionnalités de HUNCH, voir James Taggart, *Reading a Sketch by HUNCH*, MA thesis, Massachusetts Institute of Technology, Department of Electrical Engineering, 1973, 166 p.

26. Nicholas Negroponte, *Soft Architecture Machines*, op. cit., p. 65.

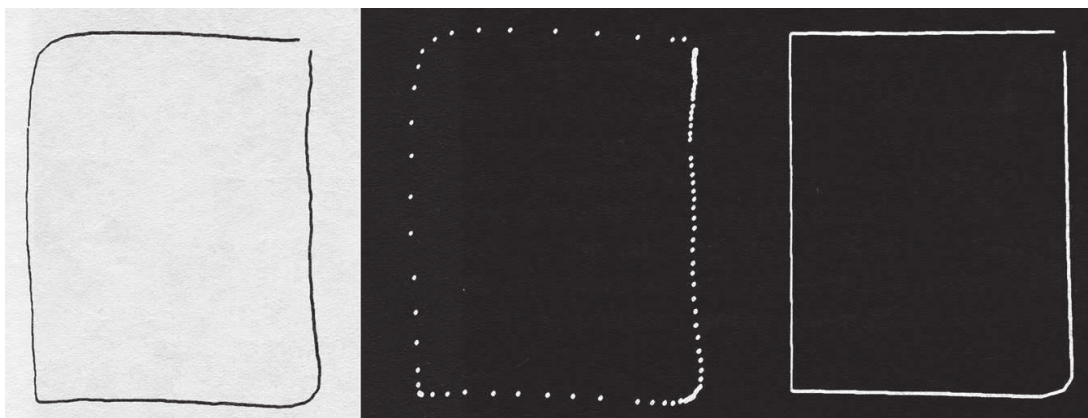
27. *Ibid.*, p. 67.



Ill. 6 : Une tablette Sylvania utilisée par un membre de l'Architecture Machine Group, c. 1970. © Cl. J. Kauffman, Image extraite de James Taggart, « Sketching : An Informal Dialogue Between Designer and Computer », Nicholas Negroponte (ed.), *Reflections on Computer Aids to Design and Architecture*, New York : Petrocelli/Charter, 1975, p. 150.

La fonctionnalité de la tablette permet également l'enregistrement de la pression – même si le logiciel n'affiche la pression qu'en termes d'épaisseur de la ligne, pas en termes d'intensité de couleur – ainsi que des données dans un champ rapproché – la tablette peut détecter le stylet à un peu plus d'un centimètre, bien que le groupe ne sache pas vraiment à quoi ceci peut lui être utile.

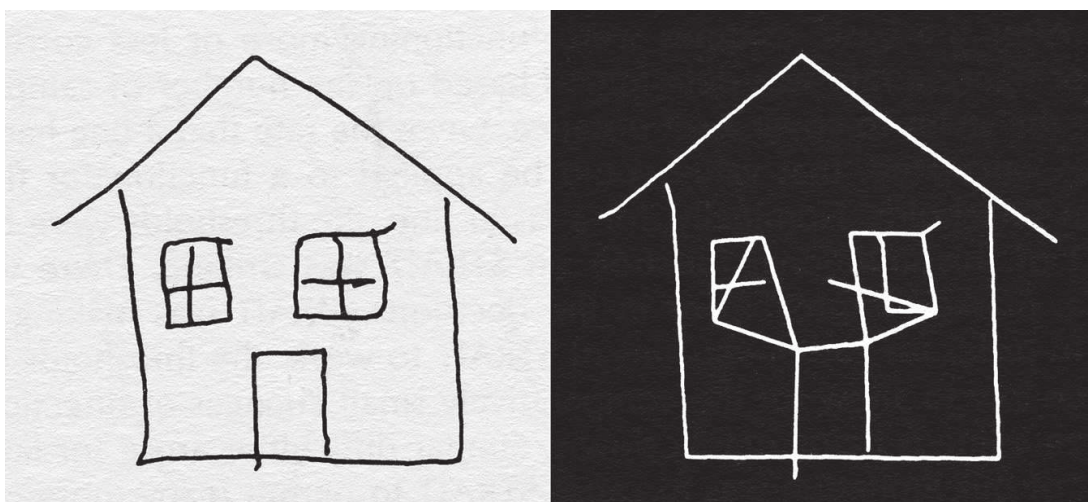
Toutes ces fonctionnalités sont programmées dans le système afin qu'il puisse, dans les faits, observer l'acte de dessiner. En enregistrant la vitesse, la pression, et l'accélération de la plume sur un morceau de papier, celui-ci est en mesure de déterminer un schéma informationnel du processus de dessin. En pratique, il est censé extraire les éléments primaires d'un dessin – les segments de ligne, les arcs, et les points – pour les rassembler ensuite en objets. L'architecte se voit alors présenter ce qui est prévu, et non pas ce qui a directement été entré. Par exemple, si une forme incomplète est dessinée, le programme la termine (ill. 7). Cependant,



Ill. 7 : Processus de reconnaissance d'esquisse. À gauche, rectangle dessiné par l'Architecture Machine Group sur une tablette Sylvania ; au milieu, illustration des points de données testés par la tablette ; à droite, interprétation par l'ordinateur de l'intention de l'utilisateur, basée sur la vitesse de ses mouvements. © Cl. J. Kauffman, Image extraite de James Taggart, « Sketching : An Informal Dialogue Between Designer and Computer », dans Nicholas Negroponte (éd.), *Reflections on Computer Aids to Design and Architecture*, New York : Petrocelli/Charter, 1975, p. 153.

comme l'expérience menée avec l'esquisse de la maison de tante Fifi et son interprétation le révèlent, il manque souvent de le faire avec précision (ill. 8).

Les limites d'un tel système sont manifestes lorsque l'on cherche à comprendre comment celui-ci pourrait être utilisé dans le processus de conception. Tout d'abord, la tablette transforme les éléments du dessin en données, et il n'est pas toujours évident de déterminer quelles données sont les plus importantes. En enregistrant seulement certains éléments d'un dessin, l'ordinateur se limite déjà systématiquement à ce que le dessin peut être et signifier, entraînant un second problème : le travail intellectuel qui s'effectue en regardant un croquis et permet de dicter les



Ill. 8 : La maison de Tante Fifi (Aunt Fifi's House). À gauche, l'image telle qu'elle est dessinée ; à droite, son interprétation par l'ordinateur. © Cl. J. Kauffman, Image extraite de James Taggart, « Sketching : An Informal Dialogue Between Designer and Computer », dans Nicholas Negroponte (éd.), *Reflections on Computer Aids to Design and Architecture*, New York : Petrocelli/Charter, 1975, p. 155.

prochaines étapes du processus se voit court-circuité, alors que par déduction l'ordinateur transforme une esquisse inachevée en quelque chose de plus abouti. Ainsi, toute signification involontaire qui pourrait résulter d'un croquis rapide sur papier n'a pas le temps de prendre forme, car le sens du dessin est dérivé d'emblée par le programme. En outre, ce sens est représenté avec une qualité toute relative : tout est retranscrit par l'ordinateur avec une précision identique. Celle-ci, trop importante pour l'esquisse, n'est toutefois pas encore assez fine pour les dessins plus élaborés. Il est ainsi impossible d'épaissir ou d'affiner les lignes, de même qu'il est impossible de modifier leur qualité de quelque façon. Le poids d'une ligne ne peut pas non plus être rendu, et plus les dessins sont complexes, plus ils sont difficiles à lire. Le sens donné à ces dessins exécutés par ordinateur se trouve donc bien plus limité que celui qui pourrait leur être attribué s'ils étaient réalisés avec un crayon et du papier.

Systèmes graphiques et conventions de dessin

Au même titre que la précision et le poids d'une ligne, il existe à cette époque d'autres éléments conventionnels du dessin que les ordinateurs ne peuvent pas encore prendre en compte. Ainsi, ni l'ombrage, ni la couleur ne peuvent être fidèlement rendus, et se trouvent seulement retranscrits à la main après une impression d'écran. Cela ne fait alors tout simplement pas partie du processus de réflexion architecturale sur ordinateur. Il faut penser au préalable à la manière de représenter son projet en fonction des éléments qui peuvent être gérés par l'ordinateur, ce qui signifie réfléchir à la conception en termes de segments de ligne et de bi-dimensionnalité. Couleurs, ombres, plans, masses et toutes les autres conventions qui s'appliquent aux représentations autres qu'en noir et blanc doivent être rejetées. En raison de ces limitations graphiques, seules quelques entités peuvent être réalisées par l'utilisateur, lesquelles sont enregistrées sous la forme de lignes solides, inflexibles et monotones.

La manière dont les systèmes informatiques attribuent du sens à ces traits dessinés force aussi le designer à travailler sur des éléments singuliers, en comprenant bien que ce qu'il dessine est seulement une suite de lignes définies par des points, des arcs, la vitesse ou l'accélération. À ce stade, les ordinateurs ne peuvent pas attribuer d'autre sens aux lignes, indiquer plus d'une proposition à la fois ou suggérer la force ou la direction. De même, ceux qui ont recours à ces systèmes informatiques doivent avoir en permanence à l'esprit quelle donnée nécessite d'être enregistrée, car les capacités de stockage des ordinateurs restent assez restreintes. Bien que celles-ci s'améliorent sensiblement à mesure que les technologies se développent, seul un nombre limité de traits par dessin peut être sauvegardé. La mémoire des ordinateurs n'étant pas extensible, l'utilisation de ces programmes doit donc être précédée d'un classement des éléments par ordre d'importance.

L'une des choses que les ordinateurs ont le plus de difficulté à prendre en compte est probablement l'un des éléments graphiques les plus simples d'un dessin

d'architecture : les lignes cachées. Il n'y a, dans cette phase précoce de développement informatique, aucun moyen de déterminer quand deux lignes dessinées se chevauchent. Les programmes ont du mal à comprendre les relations entre les deux éléments et ne peuvent donc pas prendre en compte une ligne quand elle est censée être derrière une autre. Ce qui constitue une norme graphique depuis des centaines d'années n'a pas sa place dans ces tout premiers systèmes. La reconnaissance des courbes pose également problème²⁸.

À la même période, l'architecte et théoricien Christopher Alexander – dont les *Notes on the Synthesis of Form* (1964) sont souvent invoquées comme le fondement théorique de la conception assistée par ordinateur – fait part des dangers de l'utilisation d'un système informatique qui ne peut traiter les données graphiques que de façon limitée. Ainsi, lors d'une conférence organisée en 1964 au Boston Architectural Center, Alexander souligne que l'on doit généralement faire l'effort d'indiquer un problème de façon à ce que l'ordinateur puisse être en mesure de le résoudre. L'ennui, selon lui, est que cela incite l'architecte à prendre uniquement en considération les aspects du problème qui peuvent être codés, et qui dans de nombreux cas sont « les aspects les plus futiles et les moins pertinents²⁹ ». Cette mise en garde, destinée à initier une réflexion plus large sur le processus de conception architecturale, s'applique particulièrement au dessin par ordinateur. En effet, lorsqu'un programme restreint les possibilités du dessin, il restreint également le dessin et la conceptualisation d'une idée à travers le dessin. D'un autre côté, les limitations imposées par ces systèmes pionniers favorisent des modifications et manipulations d'images qui ne sont pas aussi aisées avec des dessins traditionnels : par exemple, les ordinateurs permettent de copier et de coller des éléments de dessin beaucoup plus facilement, allant même jusqu'à supplanter les photocopieurs dans cet usage. Ils peuvent aussi zoomer et effectuer des rotations plus rapidement qu'un dessinateur ne peut le faire sur papier.

Si la représentation de la tridimensionnalité ne fait pas explicitement l'objet d'études lors de l'élaboration de ces premiers programmes, d'autres technologies sont toutefois développées en parallèle dans le but de dessiner en trois dimensions. L'une d'elles est le Lincoln Wand, élaboré en 1966 par Lawrence G. Roberts au Lincoln Laboratory du MIT³⁰ (ill. 9). Ce pointeur en forme de stylo capte des ultrasons émis par quatre transmetteurs fixés aux coins d'un écran, interpolant ainsi sa position sur les axes x et y vis-à-vis de la surface en deux dimensions de l'écran, et sur l'axe z en ce qui concerne sa distance par rapport à l'écran. Un bouton est

28. *Ibid.*, p. 77, voir la partie intitulée « Curve Recognition ». Les courbes sont initialement exclues de la programmation de Hunch jusqu'à ce qu'Arch Mac ne finisse par réaliser que les expériences menées avec ce logiciel testent souvent ces éléments afin de voir comment l'ordinateur les manipule.

29. Christopher Alexander, « A Much Asked Question about Computers and Design », in *Architecture and the Computer : Proceedings from the First Boston Architectural Center Conference, December 5, 1964 Boston, Massachusetts*, Boston, Boston Architectural Center, 1964, p. 52-56.

30. Lawrence G. Roberts, « The Lincoln WAND », *American Federation of Information Processing Societies Proceedings of the November 7-10, 1966, Fall Joint Computer Conference*, 1966, p. 223-227.



Ill. 9 : Le Lincoln Wand, c. 1966. © Photographie reproduite avec l'aimable autorisation du MIT Lincoln Laboratory, Lexington, Massachusetts.

tout ce qu'il lui faut pour enregistrer des données au sein de l'ordinateur. D'autres technologies se concentrent quant à elles sur la visualisation et la manipulation des dessins en trois dimensions, comme la balle 3-D du MIT. Ce dispositif sphérique de commande, qu'un utilisateur peut manuellement faire rouler autour de son point central, comporte en son sein des capteurs qui enregistrent ses mouvements et les transmettent au programme afin de manipuler comme tel l'image visible à l'écran. Ivan Sutherland s'implique de son côté dans la mise au point de l'un des tous premiers casques de réalité virtuelle pouvant afficher une image par rapport à la position de la tête de son porteur³¹. Par ailleurs est présenté en 1963 un petit tube cathodique conçu par R. D. Ketchpel aux Hughes Research Laboratories en Californie, qui rend possible l'affichage d'une simple petite image en trois dimensions³².

31. Ivan E. Sutherland, « A Head-Mounted Three Dimensional Display », *American Federation of Information Processing Societies Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, part I*, 1968, p. 757-766.

32. Richard D. Ketchpel, « Direct-View Three Dimensional Display Tube », *IEEE Transactions on Electron Devices*, 1963, v. 10, n° 5, p. 324-328.

Il semble néanmoins que ces expériences de la tridimensionnalité, qui auraient pu avoir des répercussions considérables sur la façon dont les concepteurs pensent et utilisent les dessins, sont bien trop éloignées des pratiques traditionnelles pour être sujettes à l'assimilation par l'industrie. C'est plutôt la souris, laquelle enregistre seulement des données en deux dimensions, qui va devenir l'interface standard de l'ordinateur pour la saisie graphique. Développée par Doug Engelbart au Stanford Research Institute entre 1961 et 1964, elle se montre extrêmement efficace et précise³³.

L'union de la technologie et de la représentation apparaît comme un élément capital pour chacun des systèmes développés au tout début des années 1960. Parce que les dessins nécessitent d'être réduits à des bits et des octets pour pouvoir être traités par les ordinateurs, on cherche à déterminer ce qui est important dans un dessin et dans la manière dont on dessine. Les technologies disponibles limitent également les façons dont les dessins peuvent être affichés – et donc lus et compris. Ceux-ci sont alors restreints à et limités par ce que les programmes informatiques sont en mesure de gérer. Objets d'une pratique établie depuis des siècles, forts de nuances et d'ambiguïtés que les ordinateurs ne peuvent reproduire, les dessins parviennent toutefois à influencer la façon dont l'informatique les assimile. Ainsi, alors que le dessin et ses pratiques s'adaptent aux ordinateurs, ces derniers, simultanément, s'ajustent aux pratiques normatives du dessin. Bien que ces premiers systèmes informatiques vont par la suite être dépassés et remplacés à mesure que les technologies se développent et que les interfaces se normalisent, c'est précisément à cette période qu'ordinateurs et dessins entrent dans une relation dialogique où chacun va influencer le développement de l'autre.

Jordan KAUFFMAN
Brandeis University (États-Unis)

33. William K. English, Douglas C. Engelbart, et Melvyn L. Berman, « Display-Selection Techniques for Text Manipulation », *IEEE Transactions on Human Factors in Electronics*, March 1967, v. HFE-8, n° 1, p. 5-15. Voir également David Evans, « Augmenting the Human Intellect », in Murray Milne (dir.), *Computer Graphics in Architecture and Design : proceedings of the Yale Conference on Computer Graphics in Architecture, held in New Haven, Connecticut, April 1968*, New Haven, Yale School of Art and Architecture, 1969, p. 61-66.